标签移动场景下的防碰撞算法研究 *

李　辽 [a]，崔晓晖 [a]，李铭哲 [b]

(武汉大学 a. 国家网络安全学院; b. 计算机学院, 武汉 430072)

摘　要：解决碰撞问题、减少标签识别时间对 RFID 的应用具有重要意义。针对目前一些已有标签防碰撞算法大多应用于标签固定场景，而在标签移动场景下表现不佳的问题，提出了一种标签移动场景下（tag moving scene，TMS）的防碰撞算法。该算法首先对移入标签和驻留标签进行区分，然后对标签数量进行预估，最后基于标签预估值采用一种混合识别策略对标签进行识别。仿真实验结果显示，相较于其他算法，TMS 算法在标签移动场景下可以有效降低标签识别时间，对 RFID 标签防碰撞算法的研究具有一定意义。

关键词：射频识别；标签防碰撞；标签预估；标签识别时间

中图分类号：TP391.44　　doi: 10.19734/j.issn.1001-3695.2018.12.0884

Research on anti-collision algorithm in tag moving scene

Li Liao[a], Cui Xiaohui[a] , Li Mingzhe[b]

(*a. School of Cyber Science & Engineering*, *b. School of Computer Science*, *Wuhan University*, *Wuhan* 430072, *China*)

Abstract: Tag collision is one of the most common and difficult problems in RFID system. Solving the collision problem and reducing the tag recognition time are of great significance for the application of RFID. In order to solve the problem that some existing anti-collision algorithms were mostly applied in tag fixed scene but had poor performance in tag moving scene, this paper proposed an anti-collision algorithm in tag moving scene (TMS) . The algorithm first distinguished between the move-in tag and the resident tag, and then estimated the number of tags, finally used a hybrid recognition strategy to identify the tag based on estimated number of tags. The simulation results show that compared with other algorithms, TMS algorithm can effectively reduce the tag recognition time in tag moving scene, which has certain significance for the research of RFID tag anti-collision algorithm.

Key words: radio frequency identification(RFID); tag anti-collision; tag estimation; tag recognition time

0　引言

RFID(radio frequency identification)是一种方便快捷的非接触式自动识别技术，它通过无线射频信号实现标签和阅读器之间的通信并识别特定目标，具有识别速度快、精度高、成本低等优点，现已在工业中得到广泛应用[1]。但当多个标签同时处于阅读器识别范围并对其响应时，会导致阅读器无法快速正确的识别标签，这种现象称为标签碰撞[2]。因此，用以解决这种标签碰撞问题的防碰撞算法研究就显得尤为重要。

受现有技术的限制，RFID 标签防碰撞方法主要分为基于树型的算法[3~9]和基于时隙随机选择的 Aloha 算法[10~16]。但是这些算法大多应用于标签固定的场景，而较少考虑标签在阅读器识别范围内移入或移出时的情况。文献[17]考虑到标签可能会移动的问题，提出了一种自适应二进制分裂算法 ABS(adaptive binary splitting)，但是该算法在驻留标签比例较高的情况下性能不佳。在此基础上，文献[18]提出了一种对分辨率阻塞算法 PRBABS（pair resolution blocking）来进一步减少 ABS 算法的标签识别时间，但是该算法在驻留标签比例较低的时候会产生大量空时隙，导致标签识别时间变长。并且这两种算法都假设驻留在阅读器识别范围内的标签数量远多于移入和移出的标签数量，这在很多情况下是不适用的，而且在标签频繁移入移出阅读器识别范围时，两种算法的标签识别时间都不是很理想，并且在标签数量的预估上会出现一定的错误。

为了解决这些问题，减少标签识别时间，本文提出了一种标签移动场景下的防碰撞算法 TMS（tag moving scene），通过在标签中记录与之通信的阅读器 ID 和当前帧 ID 来区分移入和驻留在阅读器识别范围内的标签，预估移入标签和驻留标签的数量来判断使用 ABS 或是 PRBABS 算法对标签进行识别。通过理论分析和仿真实验，本文提出的算法相较于另外两种算法，在标签移动场景下，特别是驻留标签比例较小时所需要的识别时间最少，一定程度上解决了上述问题。

1　算法描述

TMS 算法大致分为三个步骤，首先通过在标签中记录阅读器 ID 和当前帧 ID 将移入标签和驻留标签区分开来；然后对移入标签和驻留标签的数量进行预估；最后根据预估的标签数量来判别使用 ABS 算法还是 PRBABS 算法进行识别，目的是尽可能的减少标签识别时间。

1.1　基本定义

假设在一个阅读器的识别范围内不断有标签移入和移出，将阅读器识别其范围内所有标签的过程称为帧。标签选择一个时隙响应阅读器的指令并返回信息包，仅有一个标签返回

收稿日期：2018-12-19；**修回日期：**2019-02-18　**基金项目：**国家重点研发计划（2018YFC1604000）

作者简介：李辽（1996-），男，湖北武汉人，硕士研究生，主要研究方向为物联网、深度学习等；崔晓晖（1971-），男（通信作者），教授，博导，博士，主要研究方向为大数据安全等（xcui@whu.edu.cn）；李铭哲（1995-），男，江苏南通人，硕士研究生，主要研究方向为物联网，无线射频识别等.

信息包的时隙称为读取时隙，没有标签返回信息包的时隙称为空时隙，有两个或更多个标签返回信息包的时隙称为碰撞时隙。因为阅读器持续不断地识别标签，因此在识别过程中会有很多的帧产生。

为了便于算法的表达和理解，将存在于第 i 帧但不存在于第 i-1 帧中的标签定义为移入标签；将移入标签数量与第 i-1 帧中标签数量的比值定义为移入标签比例；将在第 i 帧和第 i-1 帧中都存在的标签定义为驻留标签；将驻留标签数量与第 i-1 帧中标签数量的比值定义为驻留标签比例；将存在于第 i-1 帧但不存在于第 i 帧中的标签定义为移出标签。

1.2　移入标签与驻留标签的区分机制

传统标签防碰撞算法通常只记录阅读器 ID，无法较好的将移入标签与驻留标签区分开来。TMS 算法采用的方法类似于文献[19]中提出的方法：每一个标签都有两个寄存器 A 和 B，每个阅读器有两个寄存器 A 和 C，A 用来记录标签的识别序列，B 用来记录在当前帧中已识别的标签数量，C 则用来记录所有标签里最大的寄存器 B，并且每个阅读器都有一个独一无二的 ID，标签里也有一个用来记录阅读器 ID 的变量。在每一个帧中，阅读器都存储自己的 ID 和当前帧 ID，而标签则同时存储与之通信的阅读器 ID 和当前帧 ID，并且在每个标签中都使用变量 *is_in* 来记录这是否是一个移入标签。

在每一帧的开头，阅读器都广播自己的 ID 和帧 ID，而每一个标签都将自己的阅读器 ID 和帧 ID 与之分别进行比较，若两者都相同，则判断该标签为驻留标签，并将其 *is_in* 变量置 0，否则判断其为移入标签并将 *is_in* 变量置 1。之后将阅读器和标签的帧 ID 都设置为下一帧的帧 ID，这样在下一帧中，所记录阅读器 ID 未发生改变的标签即可判断为驻留标签。通过这种方式可以将移入标签与驻留标签区分开来。

1.3　标签数量预估原则

TMS 算法通过类似于文献[20]中提出的 LOF（lottery of frame）方式来预估标签的数量。假设有 2^m 个标签，每个标签都拥有一个 m 位的二进制数 ID，一个帧包含 z 个时隙（z=1,2,3...），时隙的长短是固定的（如 0.1 秒）。以对移入标签数量进行预估的过程为例：首先阅读器在第 z 个时隙中广播数字 z 并监听标签的响应，若标签的 *is_in* 变量为 1 且标签 ID 最右边的 1 所处的低位数等于广播的数字 z，则标签作出响应。举例来说，若一个标签的 ID 为 101010，因为 ID 最右边的 1 处于低 2 位，所以当其接收到广播的数字 2 时会作出相应。这个过程会持续至第 z 个时隙中无标签响应为止。通过这种方式，在第 1 个时隙中便会有 2^{m-1} 个标签响应阅读器，第 2 个时隙中有 2^{m-2} 个标签响应阅读器...最终可用公式 $S=1.2897\times2^{m-1}$ 来预估移入标签的数量。驻留标签的数量采用同样的方法进行估计。

1.4　标签识别策略

对于 ABS 算法，在每个时隙中，只有寄存器 A 等于 B 时标签才会作出响应，并且若该时隙为读取时隙，所有的标签都会将 A 加 1；若为空时隙，所有的标签都会将 A 减 1，阅读器将 C 减 1；若为碰撞时隙，产生碰撞的标签将 A 随机增加一个二进制数，其他的标签将 A 加 1，阅读器将 C 加 1。若标签的 A 大于 B，则认为该标签是一个已经识别过的标签，当阅读器的 C 小于 A 的时候，则认为所有的标签都识别完毕，这种算法适用于驻留标签比例较低的场景。而 PRBABS 则采用类似于 TMS 的机制区分移入标签和驻留标签，在驻留标签比例较高的时候识别时间少于 ABS，但在驻留标签比例较低的时候会产生大量空时隙，导致标签识别时间变长。因此 TMS 算法采用如下标签识别策略：找到某个比例 r，当驻留标签比例小于 r 时使用 ABS 算法，否则使用 PRBABS 算法对标签进行识别。

根据文献[21]可知在 ABS 算法中预留给 m 个标签的时隙数为 0.88m，识别这些标签所需的时隙数则为 2.32m，而使用 PRBABS 算法识别标签所需的时隙为 0.5m，那么在第 i 帧中使用 ABS 算法识别驻留标签所需的时隙数为 2.32×第 i-1 帧中标签数量×驻留标签比例，PRBABS 算法识别所需时隙数为 0.5×第 i-1 帧中标签数量，则根据式子

$$2.32\times第i-1帧中标签数量\times r=0.5\times第i-1帧中标签数量 \tag{1}$$

可计算得出 r=0.215517241。根据第 1.3 节的分析，若用 z 代表阅读器的广播数字，则驻留标签比例可计算为 $1.2897\text{x}2^{z-1}$/前一帧标签总数。若驻留标签比例小于 r，则预估驻留标签数量为 $1.2897\text{x}2^{z-1}$ 个并使用 ABS 算法进行识别。而当驻留标签比例大于 r，此时由于驻留标签的数量不需要精确的估计，故阅读器预估时间可减少，这种情况下使用 PRBABS 算法。

2　算法性能分析

通过理论推导来分析计算 TMS 算法识别所有标签所需的识别时间。TMS 算法对标签的识别过程可表示为一棵二叉树的遍历，其中每个节点代表按标签识别顺序排序的标签，叶子节点代表读取时隙或者空时隙，而内部节点代表碰撞时隙。假设一棵二叉树的深度为 h，那么其节点数为 2^h。对于 m 个标签，当只有一个标签选择当前时隙，即一个叶子节点为读取时隙的时候，其概率为

$$P_{\text{read}}=C_m^1\times\left(1-\frac{1}{2h}\right)^{m-1}\times\frac{1}{2h} \tag{2}$$

当所有标签都不选择当前时隙，即一个叶子节点为空时隙的时候，其概率为

$$P_{\text{empty}}=\left(1-\frac{1}{2h}\right)^m \tag{3}$$

当一次标签碰撞发生后，一个标签上的寄存器 B 随机增加 0 或 1 的概率均为 1/2，因此在 h 次碰撞后到到达一个叶子节点的概率为

$$P_{\text{coll}}=\frac{1}{2h} \tag{4}$$

因为时隙只分为三种，故可通过反复迭代计算得到深度为 h 的二叉树中碰撞时隙的数量为

$$\begin{aligned}N_{\text{coll}}(m)&=\sum_{h=0}^{\infty}\left(1-P_{\text{coll}}-P_{\text{empty}}\right)\\&=\sum_{h=0}^{\infty}2^h\left\{1-m\times\frac{1}{2h}\times\left(1-\frac{1}{2h}\right)^{m-1}-\left(1-\frac{1}{2h}\right)^m\right\}\end{aligned} \tag{5}$$

树的节点数可计算为

$$N_{\text{node}}(m)=2\times N_{\text{coll}}(m)+1 \tag{6}$$

假设在标签移动场景下移入标签和驻留标签的数量分别为 x_1 和 y_1，TMS 算法预估的移入标签和驻留标签数量分别为 x_2 和 y_2，预估标签数量所消耗的时隙数目为 z。在驻留标签比例较小的时候，TMS 算法采用 ABS 算法进行识别，识别 x_1+y_1 个标签所消耗的时隙为 $\lceil 0.88(x_2+y_2)\rceil$。对于每一个时隙而言，在该时隙中有 a 个标签响应的概率是

$$P_{\text{reply}}=C_{x_1}^{a}\times\left(\frac{1}{\lceil 0.88x_2\rceil}\right)\times\left(1-\frac{1}{\lceil 0.88x_2\rceil}\right)^{x_1-a} \tag{7}$$

则可以计算当 a 从 0 增加到 x_1 时，识别标签消耗的时隙数为

$$N_1=\lceil 0.88(x_2+y_2)\rceil\sum_{a=0}^{x_1+y_1}C_{x_1+y_1}^{a}\left(\frac{1}{\lceil 0.88(x_2+y_2)\rceil}\right)^{a}\left(1-\frac{1}{\lceil 0.88(x_2+y_2)\rceil}\right)^{x_1+y_1-a}N_{\text{node}}(m)+z \tag{8}$$

在驻留标签比例比较大的时候，TMS 算法采用 PRBABS 算法进行识别，识别驻留标签消耗的时隙数为 $\lceil \frac{m}{2}\rceil$，识别移入标签消耗的时隙数为 $\lceil 0.88x_1\rceil$，则同样可以计算识别标签消耗的时隙数为

$$N_2=\left\lceil\frac{m}{2}\right\rceil+\lceil 0.88x_2\rceil\sum_{a=0}^{x_1}C_{x_1}^{a}\left(\frac{1}{\lceil 0.88x_2\rceil}\right)^{a}\left(1-\frac{1}{\lceil 0.88x_2\rceil}\right)^{x_1-a}N_{\text{node}}(m)+z \tag{9}$$

3　仿真实验结果与分析

下面通过计算机仿真实验验证 TMS 算法的有效性，仿真结果取相同条件下 100 次实验的平均值。实验环境设置为在第 i-1 帧中有 m 个标签需要被识别且它们都为移入标签，实验参数分别设置为：标签数量 m=100，标签 ID 长度 l=256bit，每个比特位的发送响应时间 t=5us。仿真实验首先比对 TMS 算法理论上和实际中所需要的时隙数目，然后将 ABS，PRBABS，TMS 三种算法对标签数量的预估与实际标签数量进行对比，最后计算并比较在第 i 帧里，ABS，PRBABS 和 TMS 三种算法识别完 100 个标签所花费的识别时间。

3.1　TMS 算法中理论与实际时隙数目的对比

在第 2 节中本文分析了 TMS 算法在理论上所需要的时隙数目，这里将计算的理论时隙数目与仿真实验中的实际时隙数目进行比较，目的是验证算法的理论性能分析是否正确。设置移入标签比例固定为 0.5，这样只需通过改变驻留标签比例来观察对比二者时隙数目的变化，实验结果如图 1 所示。由图可知，随着驻留标签比例由 0 到 1 增加，时隙数目的理论值和实际值都以相同的趋势变化，并且不论驻留标签比例如何改变，时隙数目理论值和实际值都十分接近，在有些地方更是完全相同（如驻留标签比例为 0.3 和 0.5 处）。由于在 TMS 算法的理论分析中计算二叉树时只是估计值，并不是准确的，因此必然会与实际值产生偏差，但根据图 1 可以计算得到二者的平均误差只有 6.73%（2.64 个时隙），由此可知理论时隙数目与实际时隙数目误差较小，即对于 TMS 算法的性能计算分析是较为正确的。

3.2　标签数量的预估

设置驻留标签比例固定为 1，这样只需通过改变移入标签比例来观察对比 ABS，PRBABS 和 TMS 三种算法的标签预估精确度。仿真实验结果如图 2 所示。由图可知，随着移入标签比例由 0 到 1 增加，实际的标签数量呈线性递增，而 TMS 算法对标签数量的预估值也几乎呈线性增长，在有些地方更是完全相同（如移入标签比例为 0.1 和 0.4 处），并且可以计算得到 TMS 预估值与实际值之间的平均误差只有 10.48%，由此可知 TMS 对标签的预估是较为准确的，这是因为 TMS 采用类似文献[23]中提出的 LOF 方式，根据当前标签的反馈来预估标签的数量。而相比于 TMS 算法，ABS 和 PRBABS 的预估值始终不变，即设置的 100 个，与实际值的平均误差达到 51.84%，无法对标签数量进行准确预估，这是由于 ABS 和 PRBABS 始终都预估标签数量为前一帧的标签数量的缘故。由以上分析可知 TMS 的标签预估精确度远高于 ABS 和 PRBABS。

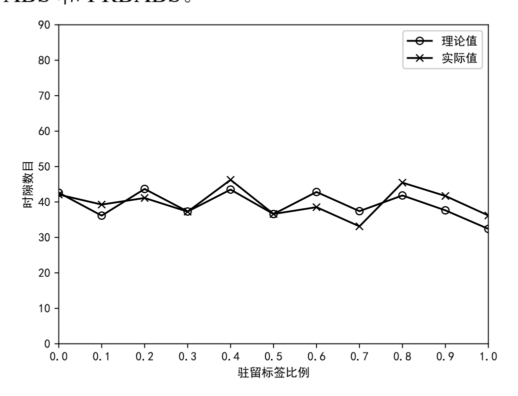


图 1　TMS 算法中时隙数目理论值与实际值的比较

Fig. 1　Comparison of theoretical and actual values of number of slots in TMS algorithm

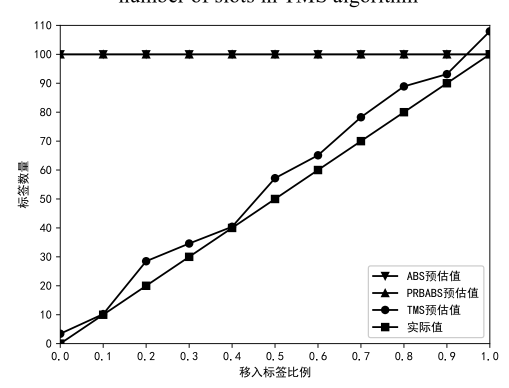


图 2　不同算法预估标签数量与实际值的比较

Fig. 2　Comparison of estimated number of labels and actual values by different algorithms

3.3　不同算法标签识别时间的对比

首先实验分析在驻留标签比例固定的情况下，三种算法在移入标签比例变化中的表现。分别设置驻留标签比例固定为 0，0.5 和 1，实验结果如图 3~5 所示。由三张图可知，随着移入标签比例由 0 到 1 改变，当驻留标签比例为 0 的时候，ABS 的标签识别时间始终要少于 PRBABS，而当驻留标签比例为 0.5 或 1 的时候，只有在移入标签比例小（0~0.2）的情况下，ABS 的表现才优于 PRBABS，这是因为 PRBABS 始终预估当前帧的移入标签数量与上一帧相同（即使是在移入标签比例小的情况下也是如此）并为它们准备时隙，所以会产生大量的空时隙，而 ABS 则是同等地识别所有标签，不会特地为移入标签准备时隙。当驻留标签比例为 0 时，TMS 在移入标签比例小于 0.6 之前效果最好，识别时间远少于另外两种算法，而当驻留标签比例为 0.5 或 1 时，TMS 在移入标签比例小于 0.4 之前效果最好，之后则都是介于 ABS 和 PRBABS 之间，这是因为相比于 ABS 和 PRBABS，TMS 对移入标签数量的预估是最精确的。在图 3 中，ABS，PRBABS 和 TMS 的平均标签识别时间分别为 74.39 个时隙，95.84 个时隙和 66.34 个时隙；在图 4 中，ABS，PRBABS 和 TMS 的平均标签识别时间分别为 107.31 个时隙，95.89 个时隙和 94.09 个时隙；在图 5 中，ABS，PRBABS 和 TMS 的平均标签识别时间分别为 139.70 个时隙，95.63 个时隙和 93.73 个时隙。因此总体来看，相对于另外两种算法，TMS 算法的平均标签识别时间最短，效果最好。

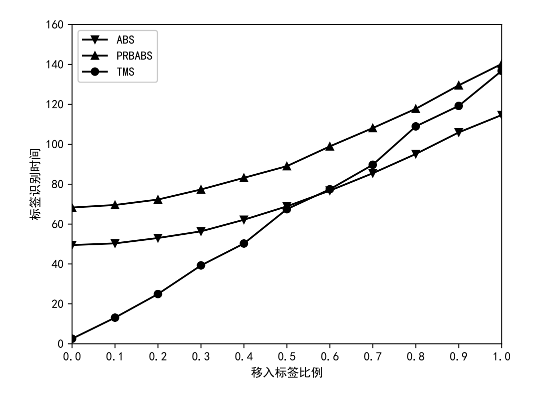


图 3 驻留标签比例=0 时三种算法的识别时间

Fig. 3 Identification time of three algorithms when the resident tag ratio=0

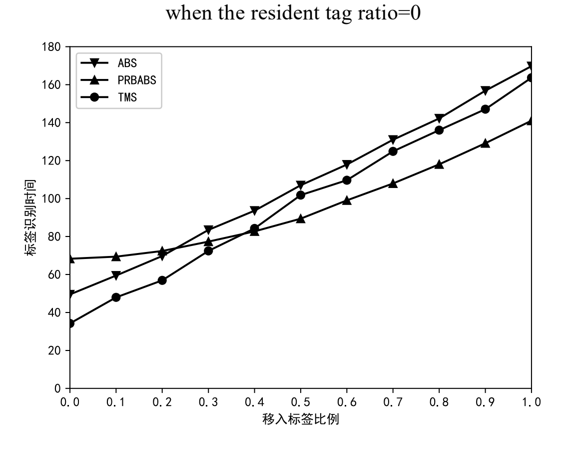


图 4 驻留标签比例=0.5 时三种算法的识别时间

Fig. 4 Identification time of three algorithms when the resident tag ratio=0.5

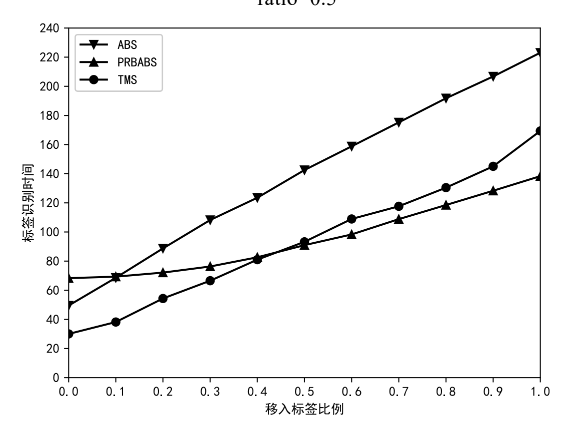


图 5 驻留标签比例=1 时三种算法的识别时间

Fig. 5 Identification time of three algorithms when the resident tag ratio=1

接着实验分析在移入标签比例固定的情况下，三种算法在驻留标签比例变化中的表现。分别设置移入标签比例固定为 0，0.5 和 1，实验结果如图 6~8 所示。由三张图可知，随着驻留标签比例由 0 到 1 改变，当移入标签比例为 0 的时候，ABS 的标签识别时间始终要少于 PRBABS，而当移入标签比例为 0.5 或 1 的时候，只有在驻留标签比例小（0~0.2）的情况下，ABS 的表现才优于 PRBABS，同样的，这也是由于 PRBABS 会始终为移入标签准备时隙而 ABS 则不会。另外还可观察到 PRBABS 的标签识别时间几乎不发生变化，这是由于 PRBABS 识别驻留标签的时候始终使用固定的时隙数目，因此驻留标签比例的改变并不会影响它的识别时间。当移入标签比例为 0 时，TMS 的效果最好，识别时间始终远少于另外两种算法，而当移入标签比例为 0.5 或 1 时，TMS 在驻留标签比例较小的时候性能是最好的。这是因为相较于另外两种算法，当驻留标签比例较小时，TMS 在识别驻留标签的过程中会产生更少的空时隙，而当移入标签比例较小时，TMS 对于移入标签数量的预估更加精确。在图 6 中，ABS，PRBABS 和 TMS 的平均标签识别时间分别为 49.50 个时隙，68.31 个时隙和 26.09 个时隙；在图 7 中，ABS，PRBABS 和 TMS 的平均标签识别时间分别为 105.88 个时隙，90.52 个时隙和 89.27 个时隙；在图 8 中，ABS，PRBABS 和 TMS 的平均标签识别时间分别为 169.64 个时隙，140.32 个时隙和 150.18 个时隙，这里 TMS 效果略差于 PRBABS 的原因是 TMS 也有着预估误差会随着移入标签比例增大而增加的缺点。不过总体来看，TMS 仍然是平均标签识别时间最短，效果最好的算法。

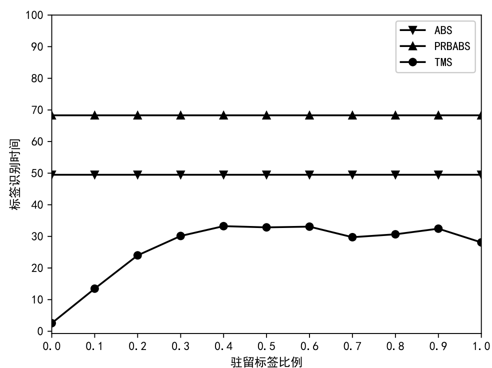


图 6 移入标签比例=0 时三种算法的识别时间

Fig. 6 Recognition time of three algorithms when moving-in tag ratio=0

图 7 移入标签比例=0.5 时三种算法的识别时间

Fig. 7 Recognition time of three algorithms when moving-in tag ratio=0.5

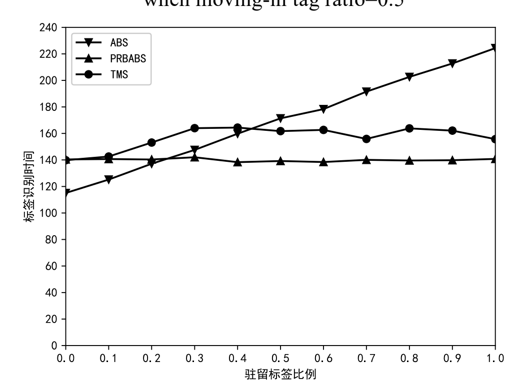


图 8 移入标签比例=1 时三种算法的识别时间

Fig. 8 Recognition time of three algorithms when moving-in tag ratio=1

4　结束语

本文针对传统标签防碰撞算法较少考虑标签移动场景的问题，通过对已有算法的分析提出了一种标签移动场景下的防碰撞算法 TMS。首先通过在标签上存储阅读器 ID 和帧 ID 来对移入标签和驻留标签进行区分，然后通过基于 LOF 的方式对移入标签和驻留标签的数量进行预估，最后根据驻留标签的比例，来决策采用 ABS 还是 PRBABS 算法对标签进行识别。理论分析和仿真实验结果表明在标签移动场景下，尤其是驻留标签比例比较小的时候，TMS 算法可以较为有效的减少识别时间。

参考文献：

[1] Lehto A, Nummela J, Ukkonen L, *et al.* Passive UHF RFID in paper industry: Challenges, benefits and the application environment [J]. IEEE Trans on Automation Science and Engineering, 2009, 6(1): 66-79

[2] 傅豪磊，陆王延. 基于多处碰撞位探测的标签防碰撞算法研究 [J]. 计算机应用研究, 2018, 35(12): 3757-3765. (Fu Haolei, Lu Wangyan. Research of tag anti-collision algorithm based on multiple collision bits detection [J]. Application Research of Computers, 2018, 35(12): 3757-3765.)

[3] Hai Linpeng，Wang Rui, Xiao Liying. A novel RFID anti-collision algorithm based on binary tree [J]. Journal of Networks, 2013, 8(12): 2885-2892.

[4] Lai Yuancheng, Hsiao Linyen, Chen HongJie, *et al.* A novel query tree protocol with bit tracking in RFID tag identification [J]. IEEE Trans on Mobile Computing, 2013, 12(10): 2063-2075.

[5] Lai Yuancheng, Hsiao Linyen, Lin Borshen. Optimal slot assignment for binary tracking tree protocol in RFID tag identification [J]. IEEE/ACM Trans on Networking, 2015, 23(1): 255-268.

[6] 刘子龙，纪金水，刘彩虹，等. 基于连续碰撞位探测的防碰撞算法研究 [J]. 电子学报，2013, 41(11): 2156-2160. (Liu Zilong, Ji Jinshui, Liu Caihong. An anti-collision algorithm based on continuous collision bit detection [J]. Acta Electronica Sinica, 2013, 41(11): 2156-2160.)

[7] 南敬昌，贾晓萌，高明明. 锁位式 RFID 自调整多叉树防碰撞算法 [J]. 计算机应用研究，2018, 35(9): 2741-2743. (Nan Jingchang, Jia Xiaomeng, Gao Mingming. Lock-bit RFID adjustive multi-tree anti-collision algorithm [J]. Application Research of Computers, 2018, 35(9): 2741-2743.)

[8] Shahzad M, Alex X L. Probabilistic optimal tree hopping for RFID identification [J]. IEEE/ACM Trans on Networking, 2015, 23(3): 796-809.

[9] Zheng Jiali, Qin Tuanfa, Ni Guangnan. Tree-based backoff protocol for fast RFID tag identification [J]. Journal of China Universities of Posts and Telecommunications, 2013, 20(2): 37-41.

[10] Wu Haifeng, Zeng Yu. Bayesian tag estimate and optimal frame length for anti-collision aloha RFID system [J]. IEEE Trans on Automation Science & Engineering, 2010, 7(4): 963-969.

[11] 李萌，钱志鸿，张旭，等. 基于时隙预测的 RFID 防碰撞 ALOHA 算法 [J]. 通信学报，2011, 32(12): 43-50. (Li Meng, Qian Zhihong, Zhang Xu, *et al.* Slot-predicting based ALOHA algorithm for RFID anti-collision [J]. Journal on Communications, 2011, 32(12): 43-50.)

[12] Qiao Juyi, Wang Weidong, Zhang Yinghai. Matching grouping dynamic Frame Slotted ALOHA for anti-collision in RFID systems [C]//Proc of International Conference on Communication Technology and Application.2011: 796-800.

[13] Solic P, Radic J, Rozic N. Early frame break policy for ALOHA-based RFID systems [J]. IEEE Trans on Automation Science & Engineering, 2016, 13 (2): 876-881.

[14] 丁治国，雷迎科. 基于优先级避让的防碰撞算法研究 [J]. 计算机应用研究, 2016, 33(3): 836-839. (Ding Zhiguo, Lei Yingkei. Research on anti-collision algorithm based on priority aversion [J]. Application Research of Computers, 2016, 33(3): 836-839.)

[15] Liu Xiulong, Li Keqiu, Wu Jie, *et al.* Top-k queries for multi-category RFID systems [C]//Proc of the 35th Annual IEEE International Conference on Computer Communications.Piscataway,Nj:IEEE Press, 2016.

[16] Chen Wentzu. Optimal frame length analysis and an efficient anti-collision algorithm with early adjustment of frame length for RFID systems [J]. IEEE Trans on Vehicular Technology, 2016, 65(5): 3342-3348.

[17] Myung J, Lee W, Srivastava J. Adaptive binary splitting for efficient RFID tag anti-collision [J]. IEEE Communications Letters, 2006, 10(3): 144-146.

[18] Lai Yuancheng, Lin Chihchung. Two blocking algorithms on adaptive binary splitting: single and pair resolutions for RFID tag identification [J]. IEEE/ACM Trans on Networking, 2009, 17(3): 962-975.

[19] Lai Yuancheng, Lin Chihchung. Two couple-resolution blocking protocols on adaptive query splitting for RFID tag identification [J]. IEEE Trans on Mobile Computing, 2012, 11(10): 1450-1463.

[20] Qian Chen, Ngan Hoilun, Liu Yunhao. Cardinality estimation for large-scale RFID systems [C]//Proc of IEEE International Conference on Pervasive Computing & Communications.Washington DC:IEEE Computer Society, 2008.

[21] Eom J B, Lee T J, Rietman R, *et al.* An efficient framed-slotted ALOHA algorithm with pilot frame and binary selection for anti-collision of RFID tags [J]. IEEE Communications Letters, 2008, 12(11): 861-863.